

WHAT IS CLAIMED IS:

5 7. A method for recovering 3D scene structure and camera motion from image data obtained from a multi-image sequence, wherein a reference image of the sequence is taken by a camera at a reference perspective and one or more successive images of the sequence are taken at one or more successive different perspectives by translating and/or rotating the camera, the method comprising the steps of:

(a) determining image data shifts for each successive image with respect to the reference image; the shifts being derived from the camera translation and/or rotation from the reference perspective to the successive different perspectives;

(b) constructing a shift data matrix that incorporates the image data shifts for each image;

(c) calculating two rank-3 factor matrices from the shift data matrix using SVD, one rank-3 factor matrix corresponding to the 3D structure and the other rank-3 factor matrix corresponding to the camera motion;

(d) recovering the 3D structure from the 3D structure matrix by solving a linear equation; and

(e) recovering the camera motion from the camera motion matrix using the recovered 3D structure.

2. The method of claim 1, wherein the image data is one or more selected from the

group consisting of points, lines and intensities.

3. The method of claim 1, wherein the step of determining image data shifts includes initially recovering and compensating for camera rotation.

4. The method of claim 1, wherein step (b) comprises:

computing H and \bar{D}_{CH} , where H is a $(N_{tot} - 3) \times N_{tot}$ matrix and $\bar{D}_{CH} \equiv C^{-1/2} \bar{D} H^T$ and $N_{tot} \equiv 2N_p + 2N_L + N_X$, where N_p , N_L , and N_X equal the number of points, lines and intensities, respectively, and C is a $(N_1 - 1) \times (N_1 - 1)$ matrix with $C_{ii'} \equiv \delta_{ii'} + 1$ where $C_{ii'}$ is a constant table of values for all images and $\delta_{ii'}$ is an operator equal to 1 if $i = i'$ and 0 if $i \neq i'$,

and $\bar{D} \equiv [S \omega_L \Lambda \omega_1 \Delta]$, H^T is the transpose of the H matrix, where H is a matrix defined such that H^T is an identity matrix and annihilates $\bar{\Psi}_x, \bar{\Psi}_y, \bar{\Psi}_z$ where $\bar{\Psi}_x^T \equiv [\Psi_x^T \omega_L \Psi_{Lx}^T \omega_1 \Psi_{Lx}^T]$ and similarly for the y and z components where ω_1 and ω_L are constant weights that the user sets, and where $\Psi_{Lx} \equiv \begin{bmatrix} \{P_U \cdot (\hat{x} \times A)\} \\ \{P_L \cdot (\hat{x} \times A)\} \end{bmatrix}$, $\Psi_{Ly} \equiv \begin{bmatrix} \{P_U \cdot (\hat{y} \times A)\} \\ \{P_L \cdot (\hat{y} \times A)\} \end{bmatrix}$, $\Psi_{Lz} \equiv \begin{bmatrix} \{P_U \cdot (\hat{z} \times A)\} \\ \{P_L \cdot (\hat{z} \times A)\} \end{bmatrix}$ where A_i is

the unit normal to the plane containing the line and the camera center at the reference image and where $\hat{x}, \hat{y}, \hat{z}$ are unit vectors in the x, y and z directions, and where

$\Psi_x \equiv \begin{bmatrix} \{r_x^{(1)}(q)\} \\ \{r_y^{(1)}(q)\} \end{bmatrix}$, $\Psi_y \equiv \begin{bmatrix} \{r_x^{(2)}(q)\} \\ \{r_y^{(2)}(q)\} \end{bmatrix}$, $\Psi_z \equiv \begin{bmatrix} \{r_x^{(3)}(q)\} \\ \{r_y^{(3)}(q)\} \end{bmatrix}$ where $q = (x, y)$ is the image position of the tracked point in the reference image and

the three point rotational flows $r^{(1)}(x, y), r^{(2)}(x, y), r^{(3)}(x, y)$ are defined by

$$[r^{(1)}, r^{(2)}, r^{(3)}] \equiv \left[\begin{pmatrix} -xy \\ -(1+y^2) \end{pmatrix}, \begin{pmatrix} 1+x^2 \\ xy \end{pmatrix}, \begin{pmatrix} -y \\ x \end{pmatrix} \right] \text{ and where}$$

$\Psi_{Lx} \equiv -\{\nabla I \cdot r^{(1)}(p)\}$, $\Psi_{Ly} \equiv -\{\nabla I \cdot r^{(2)}(p)\}$, $\Psi_{Lz} \equiv -\{\nabla I \cdot r^{(3)}(p)\}$, and where p gives the image coordinates of the pixel positions, and

where Λ is an $N_i \times 2N_L$ matrix where each row corresponds to a different image i and equals $\left[\{P_U \cdot \delta A^i\}^T \{P_L \cdot \delta A^i\}^T \right]$ where P_U and P_L are unit 3-vectors projecting onto two directions $A_i \times (\hat{z} \times A_i)$ and $\hat{z} \times A_i$, which we refer to respectively as the upper and lower

5

10

15

15

20

20

25

25

25

and where

where Q is the 3d coordinate for a 3D tracked point corresponding to an image pixel

10

$$[\overline{\Phi}_x \ \overline{\Phi}_x \ \overline{\Phi}_x] = H^T S^{(3)} U + [\overline{\Psi}_x \ \overline{\Psi}_y \ \overline{\Psi}_z] \Omega, \text{ given } U \text{ and } \Omega.$$

using $S^{(3)}U \approx [\overline{\Phi}_x \overline{\Phi}_y \overline{\Phi}_z]$ and

recovering the rotations $\omega_x^i, \omega_y^i, \omega_z^i$ from

constant $(N_l-1) \times (N_l-1)$ matrix with $C_{i''} \equiv \delta_{i''} + 1$ and T represents the translation.

g\nec\1196\13701\spec\13701pj1

reference perspective and on
or more successive different
method comprising the steps
initially recovering and compo
age by the previously estimat
termining image data shifts f
age; the shifts being derived
spective to the successive di
constructing a shift data matri
modifying the image shifts by
ese quantities to include the c
s and $1 - B \cdot T$ for lines wher
racked point, where ΔI_n^i is th
sity image and where $I_n^i = I^i$
and where δA_l^i represents th
calculating two rank-3 factor m
nk-3 factor matrix; correspon
sponding to the camera motio
covering the 3D structure fro

5

reference image; the shifts being derived from the camera translation and/or rotation from the reference perspective to the successive different perspectives;

10

15

SVD, one rank-3 factor matrix; corresponding to the 3D structure and the other rank-3 factor matrix corresponding to the camera motion;

20

equation; and

(f) recovering the camera motion from the camera motion matrix using the recovered 3D structure.

9. The method of claim 8, wherein the image data is one or more selected from the group consisting of points, lines and intensities.

10. The method of claim 8, wherein step (b) comprises:

computing H and \bar{D}_{CH} , where H is a $(N_{tot} - 3) \times N_{tot}$ matrix and $\bar{D}_{CH} \equiv C^{-1/2} \bar{D} H^T$ and $N_{tot} \equiv 2N_p + 2N_L + N_X$, where N_p , N_L , and N_X equals the number of points, lines and intensities, respectively, and C is a $(N_i - 1) \times (N_i - 1)$ matrix with $C_{ii'} \equiv \delta_{ii'} + 1$ where $C_{ii'}$ is a constant table of values for all images and $\delta_{ii'}$ is an operator equal to 1 if $i = i'$ and 0 if $i \neq i'$

and $\bar{D} \equiv [S \omega_L \Lambda \omega_I \Delta]$, H^T is the transpose of the H matrix, where H is a matrix defined such that H^T is an identity matrix and annihilates $\bar{\Psi}_x, \bar{\Psi}_y, \bar{\Psi}_z$ where $\bar{\Psi}_x^T \equiv [\Psi_x^T \omega_L \Psi_{Lx}^T \omega_I \Psi_{Ix}^T]$ and similarly for the y and z components and where ω_I and ω_L are constant weights that the

user sets, and where $\Psi_{Lx} \equiv \begin{bmatrix} \{P_U \cdot (\hat{x} \times A)\} \\ \{P_L \cdot (\hat{x} \times A)\} \end{bmatrix}$, $\Psi_{Ly} \equiv \begin{bmatrix} \{P_U \cdot (\hat{y} \times A)\} \\ \{P_L \cdot (\hat{y} \times A)\} \end{bmatrix}$, $\Psi_{Lz} \equiv \begin{bmatrix} \{P_U \cdot (\hat{z} \times A)\} \\ \{P_L \cdot (\hat{z} \times A)\} \end{bmatrix}$ where

A_i is the unit normal to the plane containing the line L and the camera center at the reference image, and where $\hat{x}, \hat{y}, \hat{z}$ are unit vectors in the x, y and z directions, and where

$\Psi_x \equiv \begin{bmatrix} \{r_x^{(1)}(q)\} \\ \{r_y^{(1)}(q)\} \end{bmatrix}$, $\Psi_y \equiv \begin{bmatrix} \{r_x^{(2)}(q)\} \\ \{r_y^{(2)}(q)\} \end{bmatrix}$, $\Psi_z \equiv \begin{bmatrix} \{r_x^{(3)}(q)\} \\ \{r_y^{(3)}(q)\} \end{bmatrix}$ where $q=(x,y)$ is the image position of the

tracked point in the reference image and

the three point rotational flows $r^{(1)}(x,y), r^{(2)}(x,y), r^{(3)}(x,y)$ are defined by

$$[r^{(1)}, r^{(2)}, r^{(3)}] \equiv \left[\begin{pmatrix} -xy \\ -(1+y^2) \end{pmatrix}, \begin{pmatrix} 1+x^2 \\ xy \end{pmatrix}, \begin{pmatrix} -y \\ x \end{pmatrix} \right] \text{ and where}$$

$\Psi_{Ix} \equiv -\{\nabla I \cdot r^{(1)}(p)\}$, $\Psi_{Iy} \equiv -\{\nabla I \cdot r^{(2)}(p)\}$, $\Psi_{Iz} \equiv -\{\nabla I \cdot r^{(3)}(p)\}$, and where p gives the image coordinates of the pixel positions, and

5

11.

15

12.

20

for the y and z components, and

25

normal to a second plane, B is defined by requiring $B \cdot A = 0$ and $B \cdot Q = -1$ for any point Q on the 3D line L,

and where $B_U \equiv B \cdot P_U$ and $B_L \equiv B \cdot P_L$ are the upper and lower components of B, and where

$$5 \quad \Phi_x \equiv -\begin{bmatrix} \{Q_z^{-1}\} \\ \{0\} \end{bmatrix}, \Phi_y \equiv -\begin{bmatrix} \{0\} \\ \{Q_z^{-1}\} \end{bmatrix}, \Phi_z \equiv -\begin{bmatrix} \{q_x Q_z^{-1}\} \\ \{q_y Q_z^{-1}\} \end{bmatrix}$$

and where

$$\Phi_{I_x} \equiv -\{Z^{-1} I_x\}, \Phi_{I_y} \equiv -\{Z^{-1} I_y\}, \Phi_{I_z} \equiv \{Z^{-1} (\nabla I \cdot p)\}, \text{ and}$$

10 where Q is the 3d coordinate for a tracked 3D point in the reference image, and Z_n is

the depth from the camera to the 3D point imaged at the n -th pixel along the cameras optical axis; and

recovering the structure unknowns Q_z , B_z and Z^{-1} from

$$[\overline{\Phi}_x \ \overline{\Phi}_y \ \overline{\Phi}_z] = H^T S^{(3)} U + [\overline{\Psi}_x \ \overline{\Psi}_y \ \overline{\Psi}_z] \Omega, \text{ given } U \text{ and } \Omega.$$

15

13. The method of claim 8, wherein step (f) comprises:

using $S^{(3)} U \approx [\overline{\Phi}_x \ \overline{\Phi}_y \ \overline{\Phi}_z]$ and

$\overline{D}_{CH} \approx C^{-1/2} \{T_x\} \overline{\Phi}_x^T H^T + C^{-1/2} \{T_y\} \overline{\Phi}_y^T H^T + C^{-1/2} \{T_z\} \overline{\Phi}_z^T H^T$ to recover the translations, and

recovering the rotations $\omega_x^i, \omega_y^i, \omega_z^i$ from

$$20 \quad \omega_x^i \overline{\Psi}_{xn} + \omega_{yx}^i \overline{\Psi}_{yn} + \omega_{zx}^i \overline{\Psi}_{zn} = C^{-1/2} \overline{D}_n^i - \left(C^{-1/2} (\{T_x\} \overline{\Phi}_x^T + \{T_y\} \overline{\Phi}_y^T + \{T_z\} \overline{\Phi}_z^T) \right)_n^i, \text{ wherein } C \text{ is a}$$

constant $(N_i - 1) \times (N_i - 1)$ matrix with $C_{ii'} \equiv \delta_{ii'} + 1$ and T represents the translation.